# Installing SLACKWARE® LINUX

Walnut Creek CDROM

**Installing Slackware Linux** by Patrick Volkerding

## TABLE OF CONTENTS


| | | | |
|---|---|---|---|
| 1.0 | Introduction | pg | 2 |
| 1.1 | Sources of Documentation | pg | 2 |
| 2.0 | Hardware Requirements | pg | 2 |
| 3.0 | Slackware Space Requirements | pg | 3 |
| 3.1 | Creating the Installation Floppies | pg | 5 |
| | Boot disc Descriptions | pg | 6 |
| 3.2 | Preparing a Partition for Slackware | pg | 14 |
| 3.3 | Using Linux fdisk to create Linux Partitions | pg | 15 |
| 4.0 | Installing the Slackware Distribution | pg | 19 |
| | Setup couldn't access your CDROM | pg | 22 |
| | MS-DOS installation | pg | 24 |
| | LILO — Linux Loader | pg | 27 |
| 4.1 | Booting the Installed Slackware System | pg | 29 |
| 4.2 | Post-installation Configuration | pg | 30 |
| | Configuring the X-Window System | pg | 30 |
| 5.0 | Using the Slackware CDROM Word Index | pg | 34 |


This document covers installation of the Slackware® distribution of the Linux operating system from the Official Slackware® CDROM.

Slackware® is a registered trademark of Walnut Creek CDROM & Patrick Volkerding
XFree86 is a trademark of The XFree86 Project, Inc.
This CDROM is copyright © 1999 Walnut Creek CDROM

**To order or to register your product:**
1-800-786-9907 • +1-925-674-0783 • Fax +1-925-674-0821
orders@cdrom.com • http://www.cdrom.com/

**For technical support:**
support@cdrom.com • +1-925-603-1234
Walnut Creek CDROM Suite 260, 1547 Palos Verdes Mall,
Walnut Creek CA 94596 USA

## 1. Introduction

Linux is a multiuser, multitasking operating system that was developed by Linus Torvalds and hundreds of volunteers around the world working over the Internet.

The Linux operating system now runs on several machine architectures, including Intel 80x86, Sparc, 68K, PowerPC, and DEC Alpha. The Slackware distribution of Linux runs on Intel (and compatible) 386, 486, Pentium, Pentium Pro, and Pentium II based PCs. Linux is modeled after the UNIX (R) operating system. The Slackware distribution contains a full program development system with support for C, C++, Fortran-77, LISP, and other languages, full TCP/IP networking with NFS, SLIP, PPP, a full implementation of the X Window System, and much more.

### 1.1. Sources of Documentation

If you're new to Slackware, you'll be happy to know there is a ***lot*** of documentation and help available both on the Internet and on the CDROM itself.

One of the primary sources of Linux information is a collection of documents known as the "Linux HOWTOs." You can find these on the Internet at metalab.unc.edu, in the /pub/Linux/docs/HOWTO directory.

Other useful documentation at that site includes:

/pub/Linux/docs/FAQ—answers to Frequently Asked Questions about Linux

/pub/Linux/docs/LDP/—Manuals written by the Linux Documentation Project.

/pub/Linux/docs/LDP/install-guide—Matt Welsh's "Linux Installation and Getting Started" book. Although the installation information contained is a little bit dated, it's still filled with useful information. If you're unfamiliar with UNIX, you'll find this guide to be a handy introduction.

Of course, these documents are also available on the first Slackware CDROM in the /docs directory, or on ftp.CDROM.com in /pub/linux/slackware/docs.

## 2. Hardware Requirements

Most PC hardware will work fine with Slackware, but some Plug-and-Play devices can be tricky to set up. In some cases you can work around this by letting DOS initialize the card and then starting Slackware with the Loadlin utility. Setting the computer's BIOS to configure Plug-and-Play cards also may help—to do this, change the "Plug and Play OS" option to "no".

Here's a basic list of what you'll need to install Slackware:

Eight megabytes (8MB) or more of RAM. If you have less than 8 megabytes, we strongly suggest adding some additional RAM. You can also try some of the procedures described in the LOWMEM.TXT file on the first CDROM. If you have eight or more megabytes of RAM, you'll be just fine.

You also will need some disk space to install Slackware. For a complete installation, you'll probably want to devote a 1GB or larger partition completely to Slackware. You can make a smaller subset of Slackware fit into as little as 100 MB, although an average installation uses around 200 to 700MB. The amount of disk space required varies dramatically depending on the amount of software you've chosen to install, the number of users on your machine, and the amount of swap space you've given Slackware. If you haven't installed Slackware before, you may have to experiment. If you've got the drive space, again, more is going to be better than not enough. Also, you can always install only the first software set (the A series containing only the basic system utilities) and then install more software later once your system is running.

If you use SCSI, Slackware supports most SCSI controllers. Check the list of bootdisks in the bootdsks.144/README.TXT file to see if there's a bootdisk for your controller. You also might consult the SCSI-HOWTO for specific information about your controller.

To install from the CDROM, you'll need a supported CDROM drive. Again,check the bootdsks.144/README.TXT file to see if your drive is listed. If it's not, you still have an excellent chance of finding a bootdisk through trial and error that works with it anyway, since many CDROM drives sold by brand-X companies contain electronics manufactured by one of the companies that make a supported drive.

Also, more and more drives made today use the ATAPI/IDE standard—these drives will all work fine under Slackware.

## 3. Slackware Space Requirements

Slackware divides the installable software into categories. (in the old days when people installed Linux from floppy disks, these were often refered to as "disk sets") Only the A series catagory (containing the base Linux OS) is mandatory, but you can't do very much on a system that only has the A series installed. Here's an overview of the software that's available for installation, along with the amount of drive space needed to install the entire set:

A The base Slackware system. Contains enough utilities to get Slackware running and have comm programs, editors, and installation utilities available. Installing the entire A series requires 40 MB.

AP Linux applications. These are some useful programs, including better editors, file quota utilities, a spell checker, man pages (and the groff package needed to process them), a Norton Commander clone called the Midnight Commander, extra shells, and other utilities. Installing the entire AP series uses 25 MB.

D Program development. This series contains compilers, interpreters, and other tools for C, C++, Objective-C, Fortran-77, Common LISP, Pascal, Perl, and more. You need this series if you plan to recompile your kernel (or anything else). Installing the entire D series will require 65 MB.

E GNU Emacs 20.3. This is a text editor with about a million extra features that allow you to read your mail, news, edit and compile programs, and just about anything else you might need to do. Installing the entire E series will require 40 MB.

F Answers to Frequently Asked Questions about Linux. This series will install useful Linux documentation, including the Linux HOWTOs, on your machine under /usr/doc/faq. Installing the F series requires about 13 MB.

K Linux kernel source. This package contains source code for the 2.2.6 Linux kernel. You'll need this (along with the C compiler and utilities from the D series) if you want to recompile your Linux kernel. Installing the K series will require 60 MB, and you'll need more to compile it.

N Networking. This package contains TCP/IP and UUCP support for Slackware, including packages to support SLIP/PPP, mail programs such as sendmail, pine, and elm, news readers like tin, trn, and nn, the Apache Web server, lynx Web browser, the netatalk Mac server, and Samba server for Windows networks. Installing the entire N series will use 35 MB.

T The teTeX TeX distribution. TeX is a typesetting language that you can use to format and print high-quality output on many types of printers. Installing the entire T series requires 95 MB.

TCL Tcl/Tk/TclX scripting languages, and tools for using Macintosh disk volumes. The TCL series needs about 14 MB.

X The X Window System, from XFree86 3.3.3.1. This series provides a system for supporting GUIs (Graphical User Interfaces) under Slackware. The entire X series requires 78 MB.

XAP Applications for the X Window System. Extra programs for X, such as file managers (TkDesk, xfm, xfilemanager), the K Desktop Environment (a full-featured, user friendly graphical user

interface), a window manager that makes X resemble Windows95 (fvwm95), graphical web browsers (Arena and Netscape Communicator), image editing and processing apps (xv, GNU gimp), a fractal generator (xfractint), communications programs, and more. Installing the entire XAP series will require about 140 MB.

XD Tools to recompile X servers. This is a kit used to relink your server, perhaps to compile in support for a new video card. Not many people will need to install this—this series is not needed to compile X applications. Installing the XD series will use about 15 MB.

XV xview3.2p1-X11R6. The XView series adds support for the Open Look window manager (commonly used on Sun systems), and for compiling XView applications. The XV series uses 11 MB.

Y The Y series contains a collection of games for Slackware. Installing the entire Y series will use about 8 MB.

You must install the A set. You probably also want to install the AP, D, and N series, as well as the X, XAP, and possibly the XV sets if you wish to run the X Window System. The Y series is fun if you have the space for it.

## 3.1 Creating the Installation Floppies

If your machine has a bootable CDROM drive (you may need to configure this in the system's BIOS settings) then you'll be able to directly boot the first CDROM. In this case, you will not need to make a pair of installation disks, although you'll still need a blank floppy disk handy to be made into your system bootdisk. If you've got a bootable CDROM drive, you can go ahead and boot the first CDROM now, and skip ahead to the section on repartitioning your hard drive.

If you don't have a bootable CDROM drive to install Slackware from, you'll need to make a set of installation floppies. These are called the "boot" and "root" disks. The bootdisk contains the Linux kernel, so you'll need to choose a bootdisk matched to your hardware. The rootdisk contains a small Linux system and the installation software. Later, you'll also need a third formatted floppy disk when the system generates a custom bootdisk to start your Slackware system.

You'll find the bootdisk images on the CDROM in the bootdsks.144 (for 1.44 MB) directory. Because they don't have enough space to support the installation system, 1.2 MB boot/root disks are not fully supported.

If you have a 1.2 MB (5.25") boot floppy drive, your options are to install a 1.44MB boot floppy drive or bootable CDROM drive (at least until the installation process is complete), or read the information in the bootdsks.12 directory for various workarounds.

**USING THE VIEW PROGRAM:** If you have MS-DOS on your machine, you can greatly simplify the process of selecting and creating your boot and root disks by running the VIEW.EXE program on the CDROM. This will let you look at the lists of boot and root disks (showing the hardware each supports) and will write out the disks automatically. To see a list of bootdisks for your machine, start VIEW.EXE under MS-DOS.

**NOTE:** Writing out the boot and root disks may not work correctly under Windows95. Restart your machine in real MS-DOS mode before attempting to create the disks.

Use the view program to switch into a directory containing Slackware bootdisks. If you have a 1.44 MB floppy drive, bootdsks.144 is the directory to use. Move the selection bar with the arrow keys until 'bootdsks.144/' is highlighted, and then hit enter to move into the directory. If you have a 1.2 MB floppy drive, move into the bootdsks.12/ directory instead. Once you're in one of the bootdsks.*/ directories, move down another level into one of the ide-bat/ or scsi-bat/ directories. These contain MS-DOS batch files to write out the bootdisks. Use the scsi-bat/ directory if your machine has a SCSI controller card, or the ide-bat/ directory if it only has IDE. You'll see a list of the bootdisks you can use. Use the view program to select one that supports the hardware you've got. You'll be prompted to insert a formatted floppy disk, and then hit a key. If all goes well, your floppy drive should start chugging away, writing the bootdisk image onto the floppy disk.

There are two main categories of Slackware bootdisks, those that support SCSI (and IDE) controllers, and those that only support IDE. The disks that only contain IDE support have a .I extension to the filename, while those with IDE and SCSI support have a .S extension. Note that ***all*** of the Slackware bootdisks have full support for IDE (including IDE/ATAPI CDROM drives). A detailed list of bootdisks for Slackware follows:

**IDE Slackware bootdisks:**

| | |
|---|---|
| aztech.i | CD-ROM drives: Aztech CDA268-01A, Orchid CD-3110, Okano/Wearnes CDD110, Conrad TXC, CyCDROM CR520, CR540. |
| bare.i | This is the disk to use for installation on most IDE based PCs. It includes support for IDE/ATAPI CD-ROM drives. Most CD-ROM drives made today fall into this category. |
| bareapm.i | This disk is the same as the bare.i image, with the addition of Advanced Power Management |

support. This is usually used on laptops to check the battery level, or to put the machine in standby or sleep mode. On machines that aren't laptops, it can cause booting problems. This is the correct disk to use for a PCMCIA install.

cdu31a.i — Sony CDU31/33a CD-ROM.

cdu535.i — Sony CDU531/535 CD-ROM.

cm206.i — Philips/LMS cm206 CD-ROM with cm260 adapter card.

goldstar.i — Goldstar R420 CD-ROM (sometimes sold in a 'Reveal Multimedia Kit').

isp16.i — Supports ISP16/MAD16/Mozart CD-ROM drives. Boot time command line options (or 'append=' options in /etc/lilo.conf) are: isp16=<port>,<irq>,<dma>,<drive_type>. Valid values for drive_type include: Sanyo, Panasonic (same as Sanyo), Sony and Mitsumi. Default values are: port=0x340, irq=0, dma=0, drive_type=Sanyo.

lowmem.i — This is a really stripped-down Linux kernel which might be useful for installing on IDE systems with a low amount of RAM (less than 8MB). If bare.i runs into problems, you might try this. NOTE: On systems with extremely low memory (4MB), ZipSlack plus the fourmeg.zip add-on (found in the zipslack directory) may boot and run even in cases where lowmem.i doesn't.

mcd.i — NON-IDE Mitsumi CD-ROM support.

mcdx.i — Improved NON-IDE Mitsumi CD-ROM support.

net.i — An IDE/ATAPI bootdisk with Ethernet support. Use this for installing over a network to an IDE hard drive.

no_kbd.i — This version of bare.i is a workaround for a problem: Some newer P2/Celeron systems have been known to reboot themselves as the kernel is starting up. This seems to happen as the keyboard LEDs are reset. The no_kbd.i does not reset the keyboard LEDs, and will boot correctly on most machines with this symptom. A BIOS upgrade from the motherboard manufacturer may fix the problem.

no_pci.i — If you're having problems with a hang during PCI probing, try this IDE-supporting bootdisk without PCI BIOS support. Some old PCI motherboards have BIOS bugs and may crash if "PCI bios support" is enabled, but they run fine without this option. In many cases the BIOS can be

fixed with a flash update from the motherboard manufacturer's web site.

optics.i — Optics Storage 8000 AT CD-ROM (the 'DOLPHIN' drive).

pportide.i — This is an extended version of bare.i will support for a wide variety of parallel-port IDE devices. Supports parallel-port products from MicroSolutions, Hewlett-Packard, SyQuest, Imation, Avatar, and other manufacturers.

sanyo.i — Sanyo CDR-H94A CD-ROM support.

sbpcd.i — Matsushita, Kotobuki, Panasonic, CreativeLabs (Sound Blaster), Longshine and Teac NON-IDE CD-ROM support. IMPORTANT! I can't possibly stress enough that this disk is ***not*** for IDE/ATAPI CD-ROM drives, which include nearly all of the drives made by these manufacturers recently. For IDE/ATAPI CD-ROM drives, use bare.i.

smp.i — This disk is the same as the generic bare.i bootdisk, but adds support for motherboards with multiple CPUs. (symmetric multiprocessing, or SMP).

xt.i — MFM (very very old) hard drive support.

**SCSI/IDE Slackware bootdisks:**

7000fast.s — Western Digital 7000FASST SCSI support.

acard.s — ACARD 870U/W SCSI host adapter support.

advansys.s — This bootdisk supports all AdvanSys SCSI controllers, including some SCSI cards included with HP CD-R/RW drives, the Iomega Jaz Jet SCSI controller, and the SCSI controller on the Iomega Buz multimedia adapter.

aha152x.s — Adaptec 152x SCSI support. This supports the AHA-1510, AHA-1520, AHA-1522, and AHA-2825 SCSI host adapters.

aha1542.s — Adaptec 1542 SCSI support.

aha1740.s — Adaptec 1740 SCSI support.

aic7xxx.s — Adaptec AIC7xxx SCSI support. These include the 274x EISA cards, 284x VLB cards, 294x PCI cards, 394x PCI cards, 3985 PCI card, and several versions of the Adaptec built-in SCSI controllers on various PC motherboards.

| | |
|---|---|
| am53c974.s | AM53/79C974 SCSI support. |
| buslogic.s | This supports BusLogic MultiMaster and FlashPoint SCSI controllers. |
| dtc3280.s | DTC (Data Technology Corp) 3180/3280 SCSI support. |
| eata_dma.s | DPT EATA-DMA SCSI support. (Boards such as PM2011, PM2021, PM2041, PM3021, PM2012B, PM2022, PM2122, PM2322, PM2042, PM3122, PM3222, PM3332, PM2024, PM2124, PM2044, PM2144, PM3224, PM3334.) |
| eata_isa.s | DPT EATA-ISA/EISA SCSI support. (Boards such as PM2011B/9X, PM2021A/9X, PM2012A, PM2012B, PM2022A/9X, PM2122A/9X, PM2322A/9X) |
| eata_pio.s | DPT EATA-PIO SCSI support. (PM2001 and PM2012A) |
| fdomain.s | This supports Future Domain's 16-bit SCSI host adapters (TMC-1660/1680, TMC-1650/1670, TMC-3260, TMC-1610M/MER/MEX) and other adapters based on the Future Domain chipsets (Quantum ISA-200S, ISA-250MG; Adaptec AHA-2920; and at least one IBM board). |
| gdt.s | This is a bootdisk for all SCSI Disk Array Controllers (EISA/ISA/PCI) manufactured by ICP Vortex. |
| ibmmca.s | This is a bootdisk based on a development kernel which supports MicroChannel Architecture, found in some IBM PS/2 machines and laptops. It is a bus system similar to PCI or ISA. Support for most MCA SCSI, Ethernet, and Token Ring adapters is included. |
| in2000.s | Always IN2000 SCSI support. |
| initio.s | Supports the Initio 91XXU(W) and Initio 91XXU(W) SCSI host adapters. |
| iomega.s | IOMEGA parallel port SCSI support. This supports the parallel-port Zip drives as a Linux SCSI device. This disk supports the older PPA parallel Zip drives as well as the newer ZipPlus drives that use the IMM driver. |
| megaraid.s | This bootdisk supports the AMI MegaRAID 428 and 438 (and maybe 466) SCSI host adapters. |
| n53c4xx.s | NCR 53c406a and Symbios Logic sym53c416 SCSI support. |
| n_5380.s | NCR 5380 and 53c400 SCSI support. This is the generic NCR family of SCSI controllers, not to |

be confused with the NCR 53c7 or 8xx controllers. An example of a controller that uses the NCR53c400 chip is the Trantor T130B.

n_53c7xx.s NCR 53c7xx, 53c8xx SCSI support. (Most NCR PCI SCSI controllers use this driver, or the driver below)

n_53c8xx.s This is the FreeBSD NCR driver adapted to Linux for the NCR53C8XX family of PCI-SCSI controllers. This driver supports parity checking, tagged command queuing, fast SCSI II transfer up to 10 MB/s with narrow SCSI devices and 20 MB/s with wide scsi devices. It also supports Ultra SCSI data transfers with NCR53C860 and NCR53C875 controllers. The following Symbios/NCR chipsets are supported by the driver in this bootdisk: 810 810A 815 825 825A 860 875 876 895 896

no_kbd.s This version of scsi.s is a workaround for a problem: Some newer P2/Celeron systems have been known to reboot themselves as the kernel is starting up. This seems to happen as the keyboard LEDs are reset. The no_kbd.s does not reset the keyboard LEDs, and will boot correctly on most machines with this symptom. A BIOS upgrade from the motherboard manufacturer may fix the problem.

no_pci.s If you're having problems with a hang during PCI probing, try this SCSI-supporting bootdisk without PCI BIOS support. Some old PCI motherboards have BIOS bugs and may crash if "PCI bios support" is enabled, but they run fine without this option. In many cases the BIOS can be fixed with a flash update from the motherboard manufacturer's web site.

old_cd.s This disk is provided in the unlikely event that someone needs to install from an old CD-ROM drive on a proprietary (°not° SCSI or IDE/ATAPI) interface to a hard drive on a SCSI controller. This disk supports most Linux supported SCSI controllers, plus the CD-ROM drives supported by these bootdisks: aztech.i, cdu31a.i, cdu535.i, cm206.i, goldstar.i, isp16.i, mcd.i, mcdx.i, optics.i, sanyo.i, and sbpcd.i.

pas16.s Pro Audio Spectrum/Studio 16 SCSI support.

pci2000.s This bootdisk supports the PCI2000I, PCI2220i, and PSI240i EIDE interface cards. All of these act as SCSI host adapters.

qlog_fas.s ISA/VLB/PCMCIA Qlogic FastSCSI! support. (also supports the Control Concepts SCSI cards

based on the Qlogic FASXXX chip)

qlog_isp.s — Supports all Qlogic PCI SCSI controllers, except the PCI-basic, which is supported by the AMD SCSI driver found on the am53c974.s bootdisk.

scsi.s — A generic SCSI bootdisk, with support for most SCSI controllers that work under Linux. (NOTE: This disk wastes a lot of memory, since it contains nearly ***all*** of the SCSI drivers. If you know which SCSI controller your system has, it's °far° better to use the disk designed especially for it. But, if you don't know, then this generic disk might just work for you.)

scsinet.s — Supports most SCSI controllers, plus many of the most common ethernet cards. Use this for installing over a network to a SCSI drive. (Or you can also use the bootdisk for your SCSI controller plus the appropriate ethernet module. See the README files in the ../modules directory if you want to try this approach). NOTE: Don't be too surprised if scsinet.s causes your machine to hang. It's loaded with nearly every SCSI and network driver for Linux, and unless your hardware is well behaved, the probing process can freeze your machine. If that happens, don't panic, just use a disk with less drivers. If you're not installing over the network, the disk designed for your SCSI controller is the right choice. If you need network support, use the disk designed for your SCSI card, plus put 'insmod' and the required network modules on another floppy to load before running setup. The README files in the../modules directory explain this process in detail.

seagate.s — Seagate ST01/ST02, Future Domain TMC-885/950 SCSI support.

sgivis.s — This bootdisk supports the SGI Visual Workstation models 320 and 540. The kernel also supports Adaptec aic7xxx SCSI cards, multiple CPUs, and the SGI DBE video frame buffer graphics console.

smp.s — This disk is the same as the generic scsi.s bootdisk, but adds support for motherboards with multiple CPUs. (symmetric multiprocessing, or SMP)

sym538xx.s — This bootdisk contains enhanced support for Symbios/NCR SCSI chipsets in the SYM53C8XX family. It drops support of some of the oldest chipsets in this family in order to use some new features, so older cards will need to use the n_53c7xx.s or n_53c8xx.s bootdisk images instead. The driver in this bootdisk supports the following Symbios/NCR chipsets: 810A 825A 860 875 876 895 896

tekram.s This bootdisk supports the Tekram DC390(T) PCI SCSI Hostadapter with the Am53C974A chip, and perhaps other cards using the same chip. It does _not_ support the DC390W/U/F adaptor with the NCR/Symbios chips -- for those cards, use the n_53c8xx.s bootdisk.

trantor.s Trantor T128/T128F/T228 SCSI support.

ultrastr.s UltraStor 14F, 24F, and 34F SCSI support.

ustor14f.s UltraStor 14F and 34F SCSI support.

You'll want to choose a bootdisk from the list that supports your installation media (such as a CDROM drive) and the hard drive you'll be installing to. For example, to install from an IDE CDROM drive to an IDE hard drive, you'd use the BARE.I disk. Or, for a system with an NCR 53c810 SCSI controller, SCSI CDROM, and SCSI hard drive, you'd use either the n_53c8xx.s or the n_53c7xx.s bootdisk.

Disks with network support are usually used to install from NFS, so you don't need to be concerned with whether the installed system will require ethernet drivers at this point. When selecting the bootdisk, you only need to think about what you need to get the system installed. You can add additional drivers after installation by recompiling the Linux kernel, or by loading the support in the form of kernel modules.

Once you've selected your bootdisk, you'll need to write it out to a floppy disk. The easiest way to do this is to use the MS-DOS VIEW.EXE program on the CDROM.

Use the view program to navigate into either the bootdsks.12 or bootdsks.144 directory. From there, move into either the SCSI-BAT or IDE-BAT directory, depending on whether you've got a SCSI controller in your system. Here, you'll see a list of options. To create a bootdisk, select one of the choices with the view program. Then follow the on-screen prompts to put a formatted floppy in your drive and write out the disk image.

You also can create the disk manually under DOS using the RAWRITE.EXE program. This utility copies a file onto a floppy disk directly, track by track. To use RAWRITE, just put a formatted high density floppy disk into your boot drive, change to the bootdsks.144 or bootdsks.12 directory on the CDROM (depending on which size disk you use), and use the RAWRITE command to write out the disk:

```
C:\> E:      (or D:, or whatever drive letter your CDROM uses)

E:\> CD BOOTDSKS.144

E:\> DIR A:
```

(NOTE: On some machines, RAWRITE.EXE will incorrectly try to use an incorrect sector size unless you make DOS look at the disk first with the DIR command.)

```
E:\BOOTDSKS.144> RAWRITE BARE.I A:
```

In the example above, you'll notice the image written out is BARE.I. You'll want to replace that with the name of the disk you've chosen to use.

Making the floppy disks from Linux or UNIX is also no problem. In most cases you can just stick a formatted floppy disk in the drive and then send the image to the drive with 'cat':

```
# cat bare.i > /dev/fd0
```

(The exact name of the floppy device varies depending on the system used)

You also can write out the disk under Linux or UNIX using the 'dd' command. This might work better in some cases than 'cat', since it ensures that the correct sector size is used:

```
# dd if=bare.i of=/dev/(rdfd0, rdf0c, fd0, or whatever) obs=18k
```

Once you've created your bootdisk, you need to pick a rootdisk. You will find these on the CDROM in the rootdsks directory. If you plan to install Slackware on its own partition, you'll probably want to use the COLOR.GZ rootdisk. If you want to install Slackware on an existing MS-DOS partition in a \LINUX directory, then you'll want to use the UMSDOS.GZ rootdisk. If you are installing to a native Linux partition through a PCMCIA device, you'll need the PCMCIA.GZ rootdisk.

**A list of Slackware rootdisks follows:**

COLOR.GZ — This is the default Slackware installation disk, used to install Slackware Linux to its own partition. The name COLOR.GZ comes from the color menus used to install Slackware.

UMSDOS.GZ — This is similar to the COLOR disk, but installs using UMSDOS—a system that allows you to install Slackware into a directory on an existing MS-DOS partition. This isn't as fast as using a native Linux filesystem, and can use more drive space because of limitations in the way DOS stores small files. The big advantage of using UMSDOS is that you can try Slackware without repartitioning your hard drive. This is much easier for the beginner, and a lot less potentially dangerous to your system.

[NOTE: If you have a bootable CDROM drive, you can try Slackware without installing anything to your hard drive simply by booting the second CDROM. You can also install a UMSDOS Linux system from within DOS or Windows by using the ZipSlack installation system in the zipslack/ directory. ]

PCMCIA.GZ — This is a version of the COLOR.GZ rootdisk with added support for PCMCIA cards. You should use this disk to install to a laptop through the network using NFS, or from a drive connected through a PCMCIA card. [NOTE: at this time there is no support for actually installing onto a PCMCIA device]

TEXT.GZ — A rootdisk that uses text-based install scripts. This is preferred by some people, but may not have as many features as the COLOR.GZ, UMSDOS.GZ, or PCMCIA.GZ rootdisks.

Once you've selected the rootdisk you want to use, you write it out to a formatted floppy -- as you did with the bootdisk. Again, you can use the VIEW program to create the rootdisk. From within the VIEW program, change into the rootdsks directory on the CDROM. Then, select an appropriate rootdisk such as COLOR.GZ, and use VIEW to select it. Place a formatted floppy disk in your drive and press enter to write the rootdisk image to the disk.

NOTE: Looking at the rootdisk images, you'll notice they end in .GZ. This stands for GNU gzip, a utility used to compress the rootdisk images. Do not uncompress the rootdisk before writing it out. The uncompressed images won't fit on floppy disks.

Now you should have two installation disks prepared, a bootdisk and a rootdisk.

If you haven't done so already, you also should format a high density diskette that you will use to make a bootdisk for your system when you are finished with the installation.

## 3.2 Preparing a Partition for Slackware

If you plan to install Slackware onto its own hard drive partition (this offers optimal performance), then you'll need to prepare one or more partitions for it. If you're planning to use the UMSDOS system, then you can skip to the next section.

A partition is a section of a hard drive that has been set aside for use by an operating system. You can have up to four primary partitions on a single hard drive. If you need more than that, you can make what is called an "extended partition." This is actually a way to make one of the primary partitions contain several sub-partitions.

Usually there won't be any free space on your hard drive. Instead, you will have already partitioned it for the use of other operating systems, such as MS-DOS or OS/2. Before you can make your Linux partitions, you'll need to remove one or more of your existing drive partitions to make room for it. Removing a partition destroys the data on it, so you'll want to back it up first.

If you've got a large DOS partition that you'd like to shrink to make space for Slackware, there's a program called FIPS in the INSTALL directory on the CDROM. This utility allows you to shrink the size of a DOS partition without destroying the data on it. The idea is to use a disk defragmenter utility to compress all of your data into the first part of the partition. MS-DOS versions 6.0 and above include DEFRAG (a reduced-feature version of Norton Speedisk) which works well for this purpose. Then you use FIPS to set a new end point for the partition, leaving free space that you can make into Linux partitions. Even though FIPS is usually safe to use, you are strongly advised to back up all of the data on the drive. (In fact, if you're unwilling to back everything up and take your chances with FIPS, you should consider purchasing Partition Magic, a commercial partition re-sizing tool: http://www.powerquest.com) You can find a detailed guide to using FIPS on the CDROM in \INSTALL\FIPS\FIPS.DOC. If you plan to use FIPS to resize your DOS partition, refer to the FIPS guide for further repartitioning instructions. To read the FIPS documentation from the MS-DOS view program, move into the install directory, then into the FIPS directory, and then select 'FIPS.DOC' to bring up the documentation on the screen.

If you plan to repartition your system manually, you'll need to back up the data on any partitions you plan to change. The usual tool for deleting/creating partitions is the FDISK program. Most PC operating systems have a version of this tool, and if you're running DOS or OS/2 it's probably best to use the repartitioning tool from that OS. Usually DOS uses the entire drive. Use DOS fdisk to delete the partition. Then create a smaller primary DOS partition, leaving enough space to install Linux (hopefully 700 MB or so, and if your machine doesn't have a lot of RAM room for another partition of 32 MB for swap space). You'll then need to reinstall DOS on your new DOS partition, and then restore your backup.

## 3.3 Using Linux fdisk to create Linux partitions

At this point, you should have a large chunk of unpartitioned space on your hard drive that you'll be making into partitions for Slackware. To do this, place the Slackware bootdisk in your machine and reboot your machine. The disk will display a screenful of information and give you the opportunity to enter extra kernel flags to help configure your hardware. Most users won't need to enter anything special here. However, if your

hardware isn't detected properly (and you're SURE the bootdisk you're using contains the correct support) then see the file BOOTING.TXT on the CDROM for instructions on using this feature. If you don't need to enter any kernel parameters (and you probably don't), then just hit enter to begin loading the kernel. You'll see a message like this displayed on the screen as the kernel loads:

```
LILO loading ramdisk
```

The kernel will then boot. As the kernel probes your machine to determine its hardware configuration, you'll see plenty of diagnostic information printed to the screen. Eventually it will display this message:

```
VFS: Insert root floppy disk to be loaded into ramdisk and press ENTER
```

Now take the bootdisk out of the drive, replace it with the rootdisk, and hit enter. Your computer will load the Slackware rootdisk into memory, and then you'll get a login prompt. Log into the system as "root".

**Creating Linux partitions**

Now you're ready to create your root Linux partition. To do this, you'll use the Linux version of fdisk. By default, Linux fdisk creates partitions on the first IDE hard drive (/dev/hda1). If you need to partition a different hard drive, you need to specify the name of the device when you start fdisk. For example:

| | |
|---|---|
| fdisk /dev/hda | (Repartition the first IDE hard drive) |
| fdisk /dev/hdb | (Repartition the second IDE hard drive) |
| fdisk /dev/sda | (Repartition the first SCSI hard drive) |
| fdisk /dev/sdb | (Repartition the second SCSI hard drive) |

(NOTE: If you prefer, you may also try a newer menu-driven version of Linux fdisk called 'cfdisk'.)

Once you've started fdisk, it will display a command prompt. First look at your existing partition table with the 'p' command:

```
Command (m for help): p

Disk /dev/sda: 255 heads, 63 sectors, 92 cylinders

Units = cylinders of 16065 * 512 bytes
```

```
Device Boot      Begin    Start      End      Blocks      Id     System
/dev/sda1        1        1          32       257008+     6      DOS 16-bit >= 32M
```

If you used FIPS to split a DOS partition, you'll see two primary DOS partitions instead of one. The second DOS partition is empty, so you'll want to go ahead and delete it to make freespace for your Linux root and swap partitions. To do this, use the 'd' command. You'll be asked which partition number you want to delete. This will probably be the second partition on the drive, but check the partition size to make sure its the right one.

Next, you'll want to use the 'n' command to create a primary partition. This will be your root Linux partition.

```
Command (m for help): n

Command action

e   extended

p   primary partition (1-4)
```

You'll want to enter 'p' to make a primary partition.

```
Partition number (1-4): 2
```

Here, you enter "2" since DOS is already using the first primary partition. Fdisk will first ask you which cylinder the partition should start on. Fdisk knows where your last partition left off and will suggest the first available cylinder on the drive as the starting point for the new partition. Go ahead and accept this value. Then, fdisk will want to know what size to make the partition. You can specify this in a couple of ways, either by entering the ending cylinder number directly, or by entering a size. In this case, we'll enter a size. To do this, you need to enter +sizeM—in this case, +450M. Here's what the screen looks like as these figures are entered:

```
First cylinder (33-92): 33

Last cylinder or +size or +sizeM or +sizeK (33-92): +450M
```

You have now created your primary Linux partition with a size of 450 MB. Next, you'll want to make a Linux swap partition. You do this the same way. First, enter another "n" to make a primary partition:

```
Command (m for help): n

Command action
```

```
e   extended

p   primary partition (1-4)
```

Enter "p" to select a primary partition. Partition 1 is in use by DOS, and you've already used partition 2 for Linux, so you'll want to enter "3" for the new partition number:

```
Partition number (1-4): 3
```

Since this is the last partition we plan to make on this hard drive, we'll specify the end cylinder manually this time. Here are the entries for this:

```
First cylinder (90-92): 90

Last cylinder or +size or +sizeM or +sizeK (33-92): 92
```

Now we need to set the type of partition to 82, used for Linux swap. The reason we didn't need to set a partition type the last time is that unless otherwise specified Linux fdisk automatically sets the type of all new partitions to 83 (Linux native). To set the partition type, use the "t" command:

```
Command (m for help): t

Partition number (1-4): 3

Hex code (type L to list codes): 82
```

Now you're ready to save the updated partition table information onto your hard drive. Use the "p" command again to check the results and be sure you're satisfied with them:

```
Command (m for help): p

Disk /dev/sda: 255 heads, 63 sectors, 92 cylinders

Units = cylinders of 16065 * 512 bytes

Device Boot    Begin    Start    End      Blocks      Id      System

/dev/sda1      1        1        32       257008+     6       DOS 16-bit >= 32M

/dev/sda2      33       33       89       465885      83      Linux native

/dev/sda3      90       90       92       16065       82      Linux swap
```

This looks good, so we'll use the "w" command to write the data out to the drive's partition table. If you want to exit without updating the partition table (if you've made a mistake), then you can exit without changing anything by using the "q" command instead.

When you exit fdisk using the "w" command, fdisk recommends that you reboot the machine to be sure that the changes you've made take effect. Unless you've created extended partitions, you can go ahead and run setup without rebooting.

Note: Sometimes fdisk will give you a message like "This drive has more than 1024 cylinders" and warn about possible problems using partitions with DOS. This is because MS-DOS suffers from a limitation that only allows access to the first 1024 cylinders on a hard drive. Since LILO (the utility used to boot Linux from a hard drive) uses the BIOS routines for disk access, it's also affected by this limitation. This means that if your drive has more than 1024 cylinders, all DOS partitions need to reside between cylinder 1 and 1024. Linux has no problem with partitions that stretch beyond cylinder 1024, but LILO may have trouble booting kernels from them. If that happens, you can still set up Loadlin to boot from your DOS partition, or use a bootdisk to start Linux.

## 4.0 Installing the Slackware distribution

You are now ready to begin installing software onto your hard drive. To start this process, enter the command "setup" and hit enter:

```
# setup
```

This starts the installation program, and puts a full-color menu on your screen with the various options needed to install Slackware. In general, you'll want to start with the ADDSWAP option. Even if you've already created and activated a swap partition manually, you'll need to run this so Slackware adds the swap partition to your /etc/fstab file. If you don't add it, your system won't use the swap space when you reboot.

Installing a typical system involves running the following options from the setup menu in this order: ADDSWAP, TARGET, SOURCE, SELECT, INSTALL, and CONFIGURE. You may also start with KEYMAP if you have a non-US keyboard layout, or with TARGET if you don't want to use a swap partition.

For the rest of this section, we'll walk through a typical installation process.

**The ADDSWAP option:**

First, we select the ADDSWAP option. The system will scan for partitions marked as type "Linux swap" and

will ask if you want to use them for swap space. Answer YES, and the system will format the partition and then make it active for swapping. Once it's finished, setup will display a message showing the line it will add to /etc/fstab to configure the swap partition at boot time. Hit enter to continue, and setup will go on to the TARGET section of the install.

NOTE: If you created a partition to use for swap space, but setup doesn't see it when it scans your drives, it's possible that the partition type hasn't been set in the partition table. Use the Linux "fdisk" program to list your partitions like this:

```
# fdisk -l

Disk /dev/sda: 255 heads, 63 sectors, 92 cylinders

Units = cylinders of 16065 * 512 bytes

Device Boot   Begin    Start    End      Blocks       Id      System

/dev/sda1     1        1        32       257008+      6       DOS 16-bit >= 32M

/dev/sda2     33       33       89       465885       83      Linux native

/dev/sda3     90       90       92       16065        83      Linux native
```

In this case, if /dev/sda3 is meant to be a Linux swap partition, you'll need to start fdisk on drive /dev/sda:

```
# fdisk /dev/sda

Command (m for help): t

Partition number (1-4): 3

Hex code (type L to list codes): 82

Command (m for help): w
```

This will change the third partition to type 82 (Linux swap) and write the partition table out to /dev/sda.

Now, when you run setup again, the ADDSWAP option should detect the Linux swap partition.

**The TARGET option:**

The next option on the setup menu is TARGET. This lets you selectwhich partition(s) you'd like to install

Slackware on, and will format them using the Linux Second Extended Filesystem. When you select the TARGET option, the system will scan for "Linux native" partitions on your hard drives. If it doesn't find any, you'll need to make sure that you've created partitions using the fdisk program, and that the partitions are labeled as type 83 (Linux native). This is the same process shown above. If you've created one or more partitions for Slackware using Linux's fdisk program then you shouldn't have any problems, since Linux fdisk (and cfdisk) sets all new partitions to type 83 (Linux native) by default.

You will see a menu listing all the Linux native partitions. Use the arrow keys to select the partition you'd like to use for your root (or primary) Linux partition and hit enter. The setup program will then ask if you'd like to format the partition. If this is a new installation of Slackware, you'll need to do this. Otherwise, if you are installing software onto an existing Linux partition, you don't need to format the partition.

There are a few options you need to know about when you format Linux partitions. First, you'll need to decide whether or not you'd like to check the partition for bad blocks when you do the format. This is usually not necessary unless you know the drive in question has problems. Checking takes a long time compared to a normal format, so you'll probably want to just go ahead and use the "Format" menu option to format the drive normally. If you have drive problems later on, then you might want to try using the "Check" option to map the bad sectors on the drive.

Once you've picked your formatting mode, setup will prompt you to provide the inode density. On Linux filesystems, entries called "inodes" store the names of files and the locations of the blocks that make up the file. You need at least one inode for each file, so if you run out of inodes then you can't make any new files, even if there seems to be plenty of space on the drive. Inodes themselves take up space on your drive, so you need to consider how you will use the partition. Usually, the default value of one inode for every 4096 bytes on the drive is fine. But if you're planning to use the drive for a news feed or something that uses many small files, then you may wish to increase the inode density to one inode for every 2048 (or even 1024) bytes. Once you've selected the inode density, setup will go ahead and format your root Linux partition. You will then return to the menu showing the partitions available for Linux.

You'll notice that the partition you just formatted is now listed as "in use." If you made some other partitions for Slackware, you'll need to go through the same process of formatting them, selecting whether or not to check for bad blocks, and setting a reasonable inode density. With these partitions there will be an additional step -- you'll need to select where you'd like to put the partition in your directory tree.

MS-DOS assigns a letter such as A:, B:, C:, etc, to each device. Unlike DOS, Linux makes your devices visible somewhere under the root directory (/). You might have /dev/hda1 for your root partition (/) and put /dev/hda2 somewhere underneath it, such as under your /home directory. When prompted for a mount location, just enter a directory such as /home, and hit enter. As you format each additional partition and place it in the filesystem tree, you'll be returned to the partition selection menu. When you've prepared all of your Linux partitions, you'll go on to the SOURCE option.

**The SOURCE option:**

The next menu option is SOURCE, where you select the physical media from which to install Slackware.

SOURCE displays a menu offering the choice of installation from CDROM, a hard drive partition, NFS, a directory (mounted manually), or floppy disks (only the A and N series are floppy-install capable). Make sure your Slackware CDROM is in your drive, and select option 1: Installing from CDROM. Next, the system will ask you if you'd like to scan for your CDROM drive or pick manually from a list. (unless you're trying to show off to your friends, go ahead and let setup scan for the CDROM drive automatically) Setup will then try to access the Slackware CDROM. If this is successful, setup will ask you what type of installation you want: "slakware" or "slaktest". We recommend the first (and default) option "slakware" for best performance. This option installs all of the Linux software to your hard drive. The other option, "slaktest", keeps the majority of the software on the CDROM. If you use the "slaktest" option, you'll have to have the second CDROM in your computer's CDROM drive whenever you run Linux, and you'll suffer a significant performance hit. On the plus side, you'll have access to a full Linux system while only using about 30 megabytes of hard drive space. Once you've selected your installation type (CDROM dependent or not) you'll go on to the SELECT section to pick which catagories of software you wish to install.

**If setup was not successful accessing your CDROM drive.**

If setup is not successful in accessing the CDROM drive, you'll need to figure out why before you can go on. The most common reason for this is that you used a bootdisk that doesn't support the CDROM drive. In this case, you'll have to figure out the correct disk to use. Refer to the list of bootdisks and the drives they support.

Another common mistake is selecting the wrong option when asked which type of CDROM drive you have. Some manufacturers make different versions of their drives, and it can be difficult to know which version you've got without trying a few different options. For example, Mitsumi makes several models of

CDROM drives; some of these come with their own interface card, and need the Linux Mitsumi driver (and are selected as a Mitsumi drive from the menu), while most of the recent Mitsumi drives plug directly onto the computer's IDE interface (and are selected as an ATAPI/IDE CDROM drive).

For ATAPI/IDE drives, you also need to know which device entry your computer uses to access the drive. Sometimes the system's hard drive will be the master device on the IDE interface, while the CDROM is connected as the slave device. In this example, the hard drive is /dev/hda, and the CDROM drive is /dev/hdb.

**Identifying your CDROM drive**

If you have no idea which device an IDE CDROM drive is connected to, you should have the system scan for it. You also can look at the messages generated by the system as it boots -- you should see a message that Slackware detected your CDROM drive along with information about what type of drive it is. You can look at these messages by using the right shift key together with the PageUp and PageDown keys to scroll the screen up and down. If you don't see a message about your drive, you're probably using the wrong bootdisk. If you're not sure which bootdisk you need to use, try these steps:

If your CDROM drive is connected to a SCSI controller, try using the SCSI.S disk. This contains most of the Linux SCSI drivers, and should identify which controller your system uses. Then make the bootdisk especially for your controller to use for the installation process.

If you have an IDE controller, keep trying the various IDE bootdisks. Watch the screen for a message saying that Slackware detected your CDROM drive.

If your machine is a little older than average, you can try the OLD_CD.S bootdisk. This contains support for a bunch of older CDROM drives that predate the ATAPI/IDE standard. Many of these drives came with a propritary interface card, or were connected to sound cards (sometimes bundled with the CDROM drive).

If Slackware still won't detect your drive, look at the file BOOTING.TXT on the CDROM. This contains a list of extra parameters you can pass to the kernel when you boot a Slackware bootdisk. These parameters can be handy to force hardware detection when the autoprobing fails. For example, you can tell the kernel to look for a Sony CDU31a drive by entering the following command line on the bootdisk's first prompt:

```
ramdisk cdu31a=0x1f88,0,PAS
```

This tells the kernel that you've got a Sony CDU31a drive connected to an interface card at address 0x1f88, interrupts are disabled, and the interface card is a Pro Audio Spectrum.

The documentation that comes with the Linux kernel also lists examples for other types of hardware. You can find these on the Slackware CDROM in the /docs/linux-2.2.6 directory (DOS may mangle this filename).

**Starting the installation from MS-DOS**

If you can't get your CDROM detected by Linux, it's possible that the kernel doesn't support it. Linux supports most hardware, but some types of hardware (most notably plug-and-play devices) can be hard to use. If you've tried everything and still can't use the CDROM drive, you can still install Slackware. You do this by copying files from the CDROM onto a DOS partition, and then installing the software from there.

The Linux Installation-HOWTO (INSTALL.TXT on the CDROM) gives an in-depth discussion of this, but here's a brief description of the steps involved:

1. Under MS-DOS, create a directory to install from, such as C:\SLACK.
2. Copy the disk sets you wish to install from the CDROM into the C:\SLACK directory with XCOPY. For instance, you'd use this command to copy the A series from the CDROM to your DOS partition from a CDROM assigned a drive letter of E:

```
XCOPY E:\SLAKWARE\A* C:\SLACK /S
```

This will copy the software into subdirectories of C:\SLACK named A1, A2, A3, etc. Do the same for any other disk sets you wish to install.

3. Run setup. When you get to the SOURCE option, tell it you want to install from a hard drive partition. Enter the source device (such as /dev/hda1) and the directory to install from (in this case, /slack) and then proceed with the installation as normal.

**The SELECT option:**

The SELECT option lets you select software to install. When you start the SELECT option, you'll see a menu where you can choose which categories of software you're interested in installing. The first series (called the A series) contains the base filesystem structure and binaries that are crucial for your system to boot and run properly. You must install the A series. Make sure that at least the selection for series A has an [X] next to it. Use the cursor keys and the space bar to select other things to install.

Once you've selected the general categories of software you wish to install, hit enter and you'll go on to the INSTALL option.

**The INSTALL option:**

This option goes through the categories of software you've chosen and installs them. The first question the INSTALL option will ask is what type of prompting you'd like to use during the installation process. A menu will show several options, including "full", "newbie", "menu", "expert", "custom", "tagpath", and "help". The help option gives detailed information on each of the choices.

Most people will want to use "full", "menu", "expert" or "newbie" mode.

The first option to consider is "full". If you select this mode, then setup assumes you want to install all the packages in each selected series and installs them all without further prompting. This is fast and easy. Of course, depending on which software catagories you've chosen, this can use a lot of drive space. If you use this option, you should be installing to a partition with at least 700 megabytes free to insure that you don't run out of drive space during the installation process.

The "newbie" mode (which was formerly known as "normal" mode) installs all of the required packages in each series. For each of the others (one by one) you'll get a menu where you can answer YES (install the package), NO (do not install the package), or SKIP (skip ahead to the next series). You'll also see a description of what the package does and how much space it will require to help you decide whether you need it or not. The "newbie" mode is verbose, requires input after each package, and can be tedious, but using it is a good way to get a basic education about what software goes into the system.

If you can decide which packages you want from less information, the "menu" option is a good choice, and goes much faster than a "newbie" mode installation. The "menu" option displays a menu before installing each series and lets you toggle items on or off with the spacebar. Certain packages that are considered crucial will be installed by default, and won't be shown on your selection menu. Once you've selected the packages you want, you hit enter and all the selected software from the series is installed automatically. This process is repeated for each software series.

The "expert" option is similar to the "menu" option, but assumes you want control over every package that could get installed. The "expert" mode lets you toggle packages individually, allowing the user to make bad decisions like turning off crucial packages or installing a package that's part of a larger set of software without installing the other parts. If you know exactly what you need, the "expert" mode offers the maximum amount of flexibility. If you don't know what you need, using the "expert" mode will allow you to install a system that's missing crucial files.

The "custom" and "tagpath" options are only used if you've created "tagfiles" for installation. In the first directory of each disk set is a file called "tagfile" containing a list of all the packages in that series, as well as a flag marking whether the package should be installed automatically, skipped, or the user should be prompted to decide. This is useful for situations where you need to install large numbers of machines (such as in a computer lab), but most users will not need to create tagfiles. If you are interested in using them, look at one of the tagfiles with an editor.

If you're new to Slackware, and you have enough drive space, you'll probably want to select the "full" option as the easiest way to install. Otherwise, the "menu" option is another good choice for most beginners. If you think you need (or would just like to see) the extra information offered by the "newbie" mode, go ahead and use that.

Once you have selected a prompting mode, the system begins the installation process. If you've chosen "menu" or "expert" mode, you'll see a menu of software to choose from right away—use the arrow keys and spacebar to pick what you need, and then hit enter to install it. If you've chosen the "newbie" mode, the installation will begin immediately, continuing until it finds optional packages. You'll get a selection menu for each of these.

If you've selected too much software, it's possible that your hard drive may run out of space during installation. If this happens, you'll know it because you'll see error messages on the screen as setup tries to install the packages. In such a case, you'll have to reinstall selecting less software. You can avoid this problem by choosing a reasonable amount of software to begin with, and installing more software later once your system is running.

Installing software on a running Slackware system is even easier than the initial installation -- just type "setup" at a prompt, select CDROM as the source, and install some more software.

Once you have installed the software on your system, you'll go on to the CONFIGURE option.

**The CONFIGURE option:**

The setup's CONFIGURE option does the basic configuration your system needs, such as creating links for your mouse and modem, setting your timezone, and more.

The CONFIGURE option will first ensure that you've installed a usable Linux kernel on your hard drive. (If you're installing from the bootable CDROM, this will be done automatically) There are a couple of generic kernels that come with the A series called IDE and SCSI, but the best kernel to install is the one from your bootdisk. To do this, select the "bootdisk" option on the kernel installation menu. The menu will prompt you to reinsert your installation bootdisk and hit enter, and then setup will copy the kernel from the bootdisk to your hard drive.

If you don't want to use that kernel, you can pick a kernel from a menu of kernels available on the CDROM, but if you install the wrong kernel the machine likely won't boot -- it's really a much better plan ro install the bootdisk kernel. Since you used it to successfully install Slackware, you know it will work on the installed system as well.

NOTE: If you install a kernel on your system that doesn't boot correctly, you can still boot your system with the installation bootdisk. To do this, you need to enter some information on the bootdisk's boot prompt. For example, if your root partition is on /dev/sda1, you'd enter this to boot your system:

```
mount root=/dev/sda1 ro
```

The "ro" option makes the root partition initially load as read-only so Linux can safely check the filesystem. If you're using the UMSDOS filesystem, use "rw" (for read-write) instead.

Once you've installed a kernel, you'll be asked if you want to make a bootdisk for your new system. This is a very good idea, so insert a formatted floppy disk and use the "lilo" option to create a bootdisk for your system.

Next, you'll be asked if you have a modem. If you do, pick the device from the list shown. This will make a link in /dev pointing to the correct device, such as /dev/modem -> /dev/ttyS1. Similarly, you'll be asked what type of mouse you have. Pick the mouse type from the menu (or hit cancel if you don't have a mouse), and setup will create a /dev/mouse link.

After this, other installation scripts will run depending on which packages you've installed. For instance, if you installed sendmail you'll be asked if you're running TCP/IP or UUCP.

### LILO

LILO is the Linux Loader, a program that allows you to boot Linux (and other operating systems) directly from your hard drive. If you installed the LILO package, you now have an opportunity to set it up.

NOTE: For UMSDOS installations, we don't recommend installing LILO. Installing LILO can be dangerous. If you make a mistake it's possible to make your hard drive unbootable. If you're new to Linux, it might be a good idea to skip LILO installation and use the bootdisk to start your system at first. You can install LILO later using the 'liloconfig' command after you've had a chance to read the information about it in /usr/lib/lilo. If you do decide to go ahead and install LILO, be sure you have a way to boot all the operating systems on your machine in case something goes wrong. If you can't boot DOS again, use the DOS command "FDISK /MBR"

to remove LILO from your master boot record.

The easiest way to set your machine up with LILO is to pick the "simple" choice on the LILO installation menu. This will examine your system and try to set up LILO to be able to boot DOS (Windows), Linux, and OS/2 partitions that it finds. If it locates the OS/2 Boot Manager, it will ask if you'd like to configure the Linux partition so that you can add it to the Boot Manager menu. (NOTE: If you use a disk overlay program for large IDE hard drives such as EZ-DRIVE, please see the warning below before installing LILO)

The "expert" option gives you much more control over the configuration of LILO. If you decide to use the "expert" option, here's how you do it. LILO uses a configuration file called /etc/lilo.conf to hold the information about your bootable partitions -- the "expert" LILO installation lets you direct the construction of this file. To create the file, first select BEGIN to enter the basic information about where to install LILO. The first menu will ask if you have extra parameters you'd like passed to the Linux kernel at boot time. If you need any extra parameters enter them here.

Next, decide where you want LILO installed. Usually you'll want to install LILO on the boot drive's MBR (master boot record). If you use a different boot manager (like the one that comes with OS/2) then you'll want to install LILO on your root Linux partition and then add that partition to the boot manager menu using its configuration tool. Under OS/2, this is the fdisk program.

NOTE: If you use the EZ-DRIVE utility (a diskmanager program supplied with some large IDE drives to make them usable with DOS) then do not install LILO to the MBR. If you do, you may disable EZ-DRIVE and render your disk unusable with DOS. Instead, install LILO to the superblock of your root Linux partition, and use fdisk to make the partition bootable. (With MS-DOS fdisk, this is called setting the "active" partition)

The next menu lets you set a delay before the system boots into the default operating system. If you're using LILO to boot more than one operating system (such as DOS and Linux) then you'll need to set a delay so you can pick which OS you'd like to boot. If you press the SHIFT key during the delay, LILO will display a prompt where you can type a label (typically DOS or Linux) to select which OS to boot. If you set the delay to 'Forever', the system will display a prompt at boot time and wait for you to enter a choice.

Next, you need to add entries for each operating system that LILO can boot. The first entry you make will be the machine's default operating system. You can add either a DOS, Linux or OS/2 partition first. For example, let's say you select "Linux". The system will display your Linux partitions and ask which one of them you'd like to boot. Enter the name (like /dev/hda1) of your root Linux partition. Then, you'll be prompted to

enter a label. This is the name you will enter at the boot time LILO prompt to select which partition you want to boot. A good choice for this is "Linux".

Adding a DOS or OS/2 partition is similar. To add a DOS partition to the LILO configuration file, select the DOS option. The system will display your DOS partitions and ask which one of them you'd like to boot with LILO. Enter the name of your primary DOS partition. Then enter a label for the partition, like "DOS". Once you've added all of your bootable partitions, install LILO by selecting the "Install" option.

**Networking**

Another configuration menu allows you to configure your machine's networking setup. First, enter a hostname for your machine. The default hostname after installation is "darkstar," but you can enter any name you like. Next, you'll be asked to provide a domain name. If you're running a stand-alone machine (possibly using a dialup link to an Internet Service Provider) then you can pick any name you like. The default domain name is "frop.org". If you are going to add the machine to a local network, you'll need to use the same domain name as the rest of the machines on your network. If you're not sure what this is, contact your network administrator for help. Once you've specified the hostname and domain name, you'll be asked if you want to set up the machine to only use loopback. If you don't have an ethernet card, just use loopback. Otherwise, say NO. Then the setup program will ask for your machine's IP address, netmask, gateway, and nameserver. Again, if you don't know what numbers you should be using, ask the person in charge of the network to help.

Once you've completed all the configuration menus, you can exit setup and reboot your machine. Simply press ctrl-alt-delete and the kernel will kill any programs that are running, unmount your filesystems and restart the machine.

## 4.1 Booting the installed Slackware system

If you've installed LILO, make sure you don't have a disk in your floppy drive -- when your machine reboots it should start LILO. Otherwise, insert the bootdisk made for your system during the configuration process and use it to boot. Also, make sure to remove the CDROM to avoid booting it, or disable your machine's CDROM booting feature in the BIOS settings.

The kernel will go through the startup process, detecting your hardware, checking your partitions and starting various processes. Eventually you'll be given a login prompt:

```
darkstar login:
```

Log into the new system as "root".

```
Welcome to Linux 2.2.6.

darkstar login: root

last login: Mon May 1 10:37:39 on ttgl

Linux 2.2.6.

You have new mail.

darkstar: ~#
```

## 4.2 Post-installation configuration

Once the system is running, most of the work is complete. However, there are still a few programs you'll need to configure. We'll cover the most important of these in this section.

**/etc/rc.d/rc.modules**

This file contains a list of Linux kernel modules. A kernel module is like a device driver under DOS. You can think of the /etc/rc.d/rc.modules file as similar to DOS's CONFIG.SYS. The file specifies which modules the system needs to load to support the machine's hardware. After booting your machine, you may find that some of your hardware isn't detected (usually an ethernet card). To provide the support, you'll need to load the correct kernel module.

To do this, edit the /etc/rc.d/rc.modules file with a text editor such as "vi", "pico", or "emacs". You'll see a list of modules, one per line. Most of these lines will have a '#' at the beginning of them which causes the line to be ignored. As an example, let's say your machine has a 3com 3c509 ethernet card. To activate support for this card, find the line with '3c509' in it, and remove the '#' from the beginning of the line -- then save the changed file. When you reboot the system, the module will load and the kernel will recognize the card.

There's a lot more information out there about kernel modules, including lists of module names and the cards they support, as well as extra options you can can add to the module lines to configure the hardware in different ways. This documentation is on the Slackware CDROM in the /docs directory. Useful files include /docs/mini/Kerneld, /docs/kernel-2.2.6/modules.txt, and /docs/kernel-2.2.6/networking/net-modules.txt.

**Configuring the X Window System**

Configuring X is a complex task. The reason for this is the vast numbers of video cards available for the PC

architecture, most of which use different programming interfaces. For this reason, you need to know quite a bit of information about your video hardware before you can configure X. Fortunately (especially since many off-the-shelf systems don't even tell you what type of card you've got) there's a utility to help. "SuperProbe" comes with XFree86 (the version of X included with Slackware) and probes your video hardware and displays as much information as it can find on the screen. To use it, just type SuperProbe at a prompt, and make a note of what type of card it says you have.

```
# SuperProbe
```

To configure X, you'll need to make an /etc/XF86Config file. This file contains lots of details about your video hardware, mouse, and monitor. It's a very complex configuration file, so fortunately the XFree86 team has written a program called "xf86config" to help with the process. To use xf86config, first start the utility from a root prompt:

```
# xf86config
```

This will present a screenful of information about xf86config. To continue, press enter. xf86config will ask you to verify you have set your PATH correctly. It should be fine, so go ahead and hit enter.

Next, select your mouse from the menu presented. If you don't see your serial mouse listed, pick the Microsoft protocol -- it's the most common and will probably work. Next xf86config will ask you about using ChordMiddle and Emulate3Buttons. You'll see these options described in detail on the screen. Use them if the middle button on your mouse doesn't work under X, or if your mouse only has two buttons (Emulate3Buttons lets you simulate the middle button by pressing both buttons simultaneously). Then, enter the name of your mouse device. The default choice, /dev/mouse, should work since the link was configured during Slackware setup. If you're running GPM (the Linux mouse server) in repeater mode, you can set your mouse type to /dev/gpmdata to have X get information about the mouse through gpm. In some cases (with busmice especially) this can work better. xf86config will ask you about enabling special key bindings. If you need this say "y". Most users can say "n" -- enter this if you're not sure.

In the next section you enter the sync range for your monitor. To start configuring your monitor, press enter. You will see a list of monitor types -- choose one of them. Be careful not to exceed the specifications of your monitor. Doing so could damage your hardware. Specify the vertical sync range for your monitor (you should find this in the manual for the monitor). xf86config will ask you to enter strings to identify the monitor type in the XF86Config file. Enter anything you like on these 3 lines (including nothing at all).

Now you have the opportunity to look at the database of video card types. You'll want to do this, so say "y", and select a card from the list shown. If you don't see your exact card, try selecting one that uses the same chipset and it will probably work fine. Then choose an X server. You should have installed the server recommended for your card, but if not, you can always go back and install that later. Choose option (5) to use the X server recommended for your video card's chipset.

When asked if you wish to set the symbolic link, say 'y', and set the link in /var/X11R6/bin. Next, tell xf86config how much RAM you have on your video card. xf86config will want you to enter some more descriptive text about your video card. If you like, you can enter descriptions on these three lines.

You'll be asked next about your RAMDAC and clock generator settings. You may enter them if you know the values, but the X server will probably successfully probe for these values. The next option is to run X -probe only to find the clock settings for the card. You can try this, and if it works it will speed up X's startup time. If it fails, it's not usually a big problem. If it causes problems with your card, don't use it.

You'll then be asked which display resolutions you want to use. Again, going with the provided defaults should be fine to start with. Later on, you can edit the /etc/XF86Config file and rearrange the modes so 1024x768 (or whatever mode you like) is the default. At this point, the xf86config program will ask if you'd like to save the current configuration file. Answer yes, and the X configuration file is saved, completing the setup process. You can start X now with the 'startx' command.

**User Accounts**

You should make a user account for yourself. Using "root" as your everyday account is considered bad form (at the very least) since you can accidently damage your system if you mistype a command. If you're logged in as a normal user, the effects of bad commands will be much more limited. Normally you'll only log in as root to perform system administration tasks, such as setting or changing the root password, installing, configuring, or removing system software, and creating or deleting user accounts.

To make an account for yourself, use the 'adduser' program. To start it, type 'adduser' at a prompt and follow the instructions. Going with the default selections for user ID, group ID, and shell should be just fine for most users.

**Passwords and security**

When choosing passwords for a Linux system that is connected to a network you should pick a strong

password. However, passwords only help protect a system from remote trespassing. It's easy to gain access to a system if someone has physical access to the console.

If you forget the root password, you can use a rescue disk to mount your root partition and edit the files containing the password information. If you have a bootable CDROM drive, the second CDROM provides a full bootable version of Linux (with no root password) and makes an excellent rescue disk. Otherwise, use the bootdisk that you used to install Linux to load the "rescue.gz" rootdisk image. At the prompt, you can manually mount the root Linux partition from your hard drive and remove the root password. For example, if your root linux partition is /dev/hda2, here are the commands to use after logging into the rescue disk as "root":

```
mount /dev/hda2 /mnt
```

```
cd /mnt/etc
```

Next, you'll need to edit the "shadow" file to remove root's password. Editors which might be available include "vi", "emacs", and "pico". "vi" and "emacs" might be more of an adventure than you need unless you've used them before. The "pico" editor is easy for beginners to use. pico shadow

At the top of the file, you'll see a line starting with root. Right after root, you'll notice the encrypted password information between two colons. Here's how root's line in /etc/shadow might look:

```
root:EnCl6vi6y2KjU:10266:0:::::
```

To remove root's password, you use the editor to erase the scrambled text between the two colons, leaving a line that looks like this:

```
root::10266:0:::::
```

Save the file and reboot the machine, and you'll be able to log in as root without a password. The first thing you should do is set a new password for root, especially if your machine is connected to a network.

**Here are some pointers on avoiding weak passwords:**

1. Never use your name (or anyone's name), birthdate, license plate, or anything relating to yourself as a password. Someone trying to break into your machine might be able to look these things up.
2. Don't use a password that is any variation of your login name.
3. Do not use words from the dictionary (especially not "password" :) or syllables of two different words

concatenated together as your password. There are automated programs floating around on the net that can try them all in a short time.

6. Do not use a number (like 123456) or a password shorter than six characters.

The strongest passwords are a mix of letters, numbers, and symbols. Here are some examples of strong passwords:

°^5g!:1? ()lsp@@9 i8#6#1°x ++c$!jke °2zt/mn1

In practice, any password containing one or two words, a number (or two), and a symbol (or two) should be quite secure.

## 5.0 Using the Slackware CDROM Word Index

The Slackware CDROM is indexed so you can quickly locate files containing any keyword. You can access this feature from the MS-DOS view program by using the F7 key. Under Linux, you can use the 'utils/lookup.lin' binary on the CDROM.

This Document Copyright 1999 Walnut Creek CDROM.

# "Neat CDROMs" — John Dvorak, PC Computing

## Linux Toolkit

Get the tools to make your Linux operating system complete! The Linux Toolkit ***six-disc set*** features Linux software from popular Internet sites like tsx-11 and sunsite.unc.edu. You get 3.6+ gigabytes of applications, source code, and development tools.

## UNIX Freeware

This disc is filled with programmer's source code to over 700 popular UNIX packages. These programs can be enhanced, modified, or compiled as is under Linux, FreeBSD, SunOS, Solaris, and other UNIX variant platforms. This is a must-have collection for all UNIX programmers!

| | |
|---|---|
| FreeBSD | Berkeley 4.4-lite BSD UNIX PC port. 100% stable, w/src. 4 disc set! |
| Complete Linux Reference | Over 1,600 pages covering installation, user admin, kernel prog. & more! |
| Linux Business Apps | Packed with productivity tools and office applications in RPM-format. |
| Linux Games++ | Contains the best games, amusements, graphics & sound apps for Linux. |
| X11R6 | Official X Consortium X Windows. Binaries for 8 systems. Src. |
| GNU | Over 75 software packages along with emacs, compilers, source & docs. |
| Perl | System management language; you get source, binaries, documents. |
| Source Code 1-3 | 3 volumes of Internet Usenet source code from the experts. |
| Webmaster's Toolkit | Graphics, utilities, docs. & source to help you create & maintain web sites. |
| Toolkit for Delphi | Over 1,500 files, including components, utilities, and tools for Delphi. |
| Hand Held Computing | Programs for today's popular PDAs like the PalmPilot, Newton, WinCE. |
| Qoole! | Make your first Quake map in 15 min. The Internet's favorite level editor! |
| TUG | Borland Turbo language User Group: mostly C & Pascal plus docs. |

These CDROMs are available at fine bookstores and software outlets everywhere. For usage instructions please see inside cover.

This CDROM copyright © 1999 Walnut Creek CDROM